When working with SQL, there are various numeric data types that can be used when defining tables and the column data types.

The exact data types (and their names, keywords) depend on the underlying RDBMS / environment that's being used. Hence the official docs are always a great place to go.

- MySQL Data Types Documentation: https://dev.mysql.com/doc/refman/8.0/en/data-types.html
- Postgresql Data Types Documentation: https://www.postgresql.org/docs/current/datatype.html

When it comes to **numeric data types**, the amount of available data types can be confusing. Besides the differentiation between integers ("whole numbers") and approximated + exact numbers with decimal places, there also are differences within these main groups to allow for different value ranges (e.g. integers between -128 and 127 vs integers between way larger max and min numbers).

Those "size categories" exist because different value types occupy different amounts of storage (and memory) space and can affect performance. So choosing the proper category can be important to optimize the disk storage and memory usage (as well as performance).

Below, you find an overview of all the numeric value types as well as links to the official docs that summarize the available numeric value types as well as their value ranges:

MySQL

Supported integer data types:

- INT (or INTEGER)
- TINYINT
- SMALLINT
- MEDIUMINT
- BIGINT

For a comparison (e.g. min / max values), see this table in the official docs: https://dev.mysql.com/doc/refman/8.0/en/integer-types.html

Supported fixed point data types (exact numbers with decimal values):

- DECIMAL
- NUMERIC

For MySQL, these two value types are equal.

Official docs on NUMERIC / DECIMAL: https://dev.mysql.com/doc/refman/8.0/en/fixed-point-types.html

Supported floating point data types (approximate numbers with decimal values):

- FLOAT
- DOUBLE

DOUBLE offers twice the precision (i.e. the number of digits that are stored approximately after the decimal point) than FLOAT.

For more details, see the official docs: https://dev.mysql.com/doc/refman/8.0/en/floating-point-types.html

Postgresql

An overview of ALL numeric values can be found here: https://www.postgresql.org/docs/current/datatype-numeric.html

Supported integer data types:

- INT (or INTEGER)
- SMALLINT
- BIGINT

Supported fixed point data types (exact numbers with decimal values):

- DECIMAL
- NUMERIC

For Postgresql, these two value types are equal *(just as for MySQL)*.

Official docs on NUMERIC / DECIMAL: https://www.postgresql.org/docs/current/datatype-numeric.html#DATATYPE-NUMERIC-DECIMAL

Supported floating point data types (approximate numbers with decimal values):

- REAL
- DOUBLE PRECISION
- FLOAT (the same as REAL)

DOUBLE PRECISION offers twice the precision (i.e. the number of digits that are stored approximately after the decimal point) than REAL.

For more details, see the official docs:https://www.postgresql.org/docs/current/datatype-numeric.html#DATATYPE-FLOAT